Chapter 17

Algorithmic Game Theory

Computer science met game theory when the internet became a system of interacting selfish agents—routers, bidders, miners, platforms. Three founding questions organize the field (Nisan–Roughgarden–Tardos–Vazirani): *Can equilibria be computed efficiently? How bad is selfish behavior? How should we design mechanisms under computational constraints?*

17.1 The complexity of computing Nash equilibria

Zero-sum games are solvable in polynomial time via linear programming (Chapter 12). General games appear fundamentally harder.

Definition 17.1: The class PPAD (Papadimitriou 1994)

PPAD ("Polynomial Parity Argument, Directed") is the class of total search problems reducible to END-OF-LINE: given a concisely described directed graph where every node has in- and out-degree at most one, and given one unbalanced node, find another. Existence is guaranteed by a parity argument—just as Nash equilibria are guaranteed by a fixed-point theorem—but no efficient algorithm for actually *finding* the object is known.

Theorem 17.1: Daskalakis–Goldberg–Papadimitriou (2006); Chen–Deng (2006)

Computing a Nash equilibrium of a finite game is PPAD-complete, even for two-player games and even for any polynomial approximation precision. Under the widely believed conjecture PPAD $\neq$ P, there is no polynomial-time algorithm for Nash equilibrium.

Key Idea: A critique of the concept itself

"If your laptop can't find it, neither can the market" (Kamal Jain). The hardness result is not merely about algorithms: it questions Nash equilibrium as a *prediction* for large systems, since no plausible adaptive process can reach it quickly in the worst case. It redirects attention to tractable alternatives—correlated equilibria and the outcomes of learning dynamics—and to structured subclasses (zero-sum, potential games) where computation is easy.

Related results: finding Nash equilibria with specific properties (e.g., maximizing welfare) is NP-hard; the Lemke–Howson pivoting algorithm (the field's simplex-method analogue) has exponential worst cases yet works well in practice.

17.2 Potential games and best-response dynamics

Definition 17.2: Potential game (Monderer–Shapley 1996)

G is an *exact potential game* if there exists $\Phi : A \to \mathbb{R}$ with

$$u_i(a_i', a_{-i}) - u_i(a_i, a_{-i}) \;=\; \Phi(a_i', a_{-i}) - \Phi(a_i, a_{-i})$$

for every player i and all actions: every unilateral deviation changes a single global function by exactly the deviator's gain.

Theorem 17.2: Properties of potential games

Every finite potential game has a pure-strategy Nash equilibrium (any maximizer of Φ), and best-response dynamics converges to one in finite time (each step strictly increases Φ). Congestion games (Rosenthal 1973)—players choose resource subsets, each resource's cost depends on its load—are exactly the finite potential games, with $\Phi(a) = \sum_r \sum_{k=1}^{\ell_r(a)} c_r(k)$.

Traffic routing, network formation, and load balancing are congestion games: this is why decentralized traffic reliably *settles* (an equilibrium selection dynamic that provably works), even though finding the equilibrium a system will settle on can still be hard (PLS-complete).

17.3 The price of anarchy

Definition 17.3: Price of anarchy (Koutsoupias–Papadimitriou 1999)

For a game with social cost function C,

$$\mathrm{PoA} \;=\; \frac{\max_{\text{equilibria } a} C(a)}{\min_{a'} C(a')} \,,$$

the worst-case ratio of equilibrium cost to optimal cost.

Theorem 17.3: Selfish routing (Roughgarden–Tardos 2002)

In nonatomic selfish routing with affine cost functions, the price of anarchy is at most $\frac{4}{3}$—on *every* network topology, with the Pigou example (Example 7.3) attaining the bound. With polynomial costs of degree d, the bound grows as $\Theta(d/\ln d)$.

Remark: Smoothness and robustness

Roughgarden's *smoothness* framework shows that most known PoA bounds automatically extend from Nash equilibria to coarse correlated equilibria—hence to the long-run outcomes of arbitrary no-regret learners. The pessimistic worst case over a hard-to-reach concept becomes a robust guarantee about plausible adaptive behavior: one of the field's deepest ideas.

17.4 Learning dynamics and correlated equilibria

Definition 17.4: External regret; coarse correlated equilibrium

An online algorithm choosing mixed actions σ^t against payoff sequences has *external regret* $R_T = \max_a \sum_{t \le T} u(a, a^t_{-i}) - \sum_{t \le T} u(\sigma^t, a^t_{-i})$; it is *no-regret* if $R_T/T \to 0$ against any opponent. A distribution μ over action profiles is a *coarse correlated equilibrium* (CCE) if no player gains by a constant unilateral deviation from a profile drawn from μ; a *correlated equilibrium* (CE, Aumann 1974) if no player gains by any *swap function* applied to her recommended action. Nash $\subseteq$ CE $\subseteq$ CCE.

Theorem 17.4: Learning converges to equilibrium sets

(i) The multiplicative weights algorithm achieves regret $O(\sqrt{T \log |A|})$. (ii) If all players use no-external-regret algorithms, the empirical distribution of play converges to the set of CCE. (iii) If all use no-*swap*-regret algorithms (Foster–Vohra; Hart–Mas-Colell's regret matching; Blum–Mansour), it converges to the set of *correlated* equilibria. (iv) In two-player zero-sum games, the marginal empirical frequencies converge to minimax-optimal strategies.

Correlated equilibria are also computable in polynomial time by linear programming, even in succinct multiplayer games—the tractable face of equilibrium theory. This cluster of results powers practice: counterfactual regret minimization solved poker (Chapter 12); self-play learning underlies AlphaZero-style systems; and ad platforms analyze bidder learning dynamics rather than assuming instant equilibrium.

17.5 Computational mechanism design and Bitcoin's game theory

- **Combinatorial auctions**: welfare maximization is NP-hard, and VCG with approximate optimization loses truthfulness; designing mechanisms that are simultaneously truthful and computationally feasible is a rich frontier (maximal-in-range mechanisms, posted prices).

- **Simple vs. optimal**: real ad markets abandoned exact optimality for robust simplicity. Landmark bounds: a single second-price auction with an extra bidder beats Myerson (Bulow–Klemperer, Example 14.1); simultaneous first-price auctions for many items achieve a constant fraction of optimal welfare at equilibrium (smoothness again); posted prices with the right thresholds achieve half of optimal (prophet inequalities)—the theory behind "buy-it-now" pricing.

- **Incentives in protocols**: Bitcoin's security is a game-theoretic, not purely cryptographic, claim—honest mining is (approximately) an equilibrium only under bounds on miner size; *selfish mining* (Eyal–Sirer 2014) is profitable for pools above about a quarter of total hash power. Transaction fee mechanisms (EIP-1559), maximal extractable value, and validator incentives are live mechanism-design problems.

Exercises

Exercise 17.1: A potential for Cournot

Show the two-firm Cournot game with linear demand (Example 4.2) is an (ordinal) potential game with $\Phi(q_1, q_2) = q_1 q_2 \cdot (-1) + \dots$ — construct an exact potential explicitly, verify the defining identity, and use it to re-derive the equilibrium.

Exercise 17.2: Price of anarchy in Pigou networks

Generalize Example 7.3: road 2 has cost x^d. Compute equilibrium and optimal flows and show PoA $\to \infty$ as $d \to \infty$, at rate $\Theta(d/\ln d)$.

Exercise 17.3: CCE that is not CE

Exhibit a small game and a distribution over its action profiles that is a coarse correlated equilibrium but not a correlated equilibrium, and identify the profitable swap deviation.

Exercise 17.4: Load balancing

n jobs (players) each choose one of 2 identical machines; a job's cost is the number of jobs on its machine. Show this is a congestion game, compute its exact potential, find all pure equilibria, and bound the price of anarchy for the makespan (maximum load) objective.